SOUL System
Secure Online USB Login System
Everything is going online

- Social Interactions
- Banking
- Transactions
- Meetings
- Businesses

... including *all sorts of crimes* and even *war*
Our online identities

Our IDENTITY = Our PASSWORD
What if your password gets stolen?

- Identity Theft
- Money Loss
- Data Loss
- Privacy Problems

Our PASSWORD = $
<table>
<thead>
<tr>
<th>Technology</th>
<th>Problems</th>
</tr>
</thead>
</table>
| **https://** | **Security**: Prone to keylogger and brute force attacks  
**Cost**: SSL Certificates cost a lot of money |
| | **Practicality**: Requires specialized hardware token  
**Cost**: Hardware component alone will cost money  
**Visibility**: Immediately recognizable security token |
| | **Practicality**: Requires specialized hardware devices  
**Cost**: Hardware component alone will cost money  
**Consistency**: Never 100% accurate and foolproof |
Our Solution – SOUL System

- Create a two-factor authentication system that converts an ordinary hardware token (e.g. USB Flash drive) into a security token

Password + Ordinary Hardware Token

- Secure
- Low-cost
- Practical
- Invisible
- Portable
- Flexible
- Consistent
Our Solution – SOUL System

- The SOUL System aims to secure multiple websites all at once by providing a Software Development Kit and a Trusted Third Party for easy integration and registration.
Our Solution – SOUL System

1) Website uses **Software Development Kit** to integrate existing website with the SOUL System

   WEBSITE + SOUL SDK → SOUL INTEGRATED WEBSITE

2) Website registers to the **Trusted Third Party** to allow TWO-FACTOR login (e.g. USB secure login)

   INITIAL REGISTRATION → SOUL INTEGRATED WEBSITE

   TRUSTED THIRD PARTY
Our Solution – SOUL System

1) User registers an ordinary digital device such as USB Flash drive in the Trusted Third Party in order to have a SOUL Account.

2) Registered and processed login devices can now be used to register and login to SOUL Integrated Websites.
Design Challenges

- System should work in major operating systems.
- System can easily be integrated with any existing website.
- System must not require specialized hardware.
- System must be able to handle lost, stolen, or corrupted physical passwords or keys.
- System must work with very minimal installation.
Authentication Flow

1. User mounts SOUL token
2. User opens website and finds the embedded SOUL Plugin
3. User selects the image where the encrypted data is hidden and the password is typed.
4. User is signed in to the website
What Makes it Different?

- “Plug and Play” – Website integrates the SOUL System and registers to the Trusted Third Party to allow secure login
- **Low-Cost and low-maintenance** - No specialized hardware devices and system relies heavily on program codes
- **Portable to website users** – No operating system restriction and nothing installed in login devices
- **Extremely flexible** – The design of the system can be modified to fit the needs of the business
- **It’s secure and it’s a champ** – Kaspersky International Cup 2012 and Kaspersky Asia Pacific & MEA Cup Winning Research Paper
Secure Storage

- Steganography

Trick: Hide encrypted data inside images!
Result: Secure + Invisible. Ordinary USB Flash drive containing image still looks ordinary!
Secure Data Transfer

HYBRID CRYPTOSYSTEM

SIGNED JAVA APPLET

ENCRYPTION + STEGANOGRAPHY INSIDE IMAGE FILES
Implementation

1. XLCrypt and SOUL System SDK
   - Java / Python / PHP
   - RSA, AES, SHA-512, and other fxns

2. Signed Java Applet
   - Embedded in website
   - Has local filesystem access

3. Trusted Third Party
   - Primarily acts as storage of public keys & file hash values of image files
Fighting against known attacks

- Keylogging attack
- Brute-force attack
- Collision attack
- Dictionary attack
- Man-in-the-middle attack
- Reply attack
- Cloning attack
<table>
<thead>
<tr>
<th>Objective</th>
<th>Results and Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security</td>
<td>System has been secured with hybrid cryptosystem and other security features such as UUIDs, Message UUIDs, RSA Signing and Verification, double password hashing</td>
</tr>
<tr>
<td>Cost</td>
<td>Low-cost: No specific hardware components required to use the system</td>
</tr>
<tr>
<td>Portability</td>
<td>No programs are installed inside security tokens. Any hardware or digital container can be used (laptops, USB flash drives, cellphones, dropbox containers)</td>
</tr>
<tr>
<td>Flexibility</td>
<td>System currently supports Java, Python, and PHP websites. The protocol and mechanisms proposed in the system can support any language (e.g. Ruby).</td>
</tr>
<tr>
<td>Visibility</td>
<td>Data is encrypted and then stored inside image files. No programs are installed inside the security tokens.</td>
</tr>
<tr>
<td>Practicality</td>
<td>Backup key system, password change possibility even with 2 keys, additional security options because of flexibility of usage (laptops as security tokens, dropbox storage as security tokens, cellular phones as security tokens)</td>
</tr>
</tbody>
</table>
The End